FCNI Survey Cloud Infrastructure
How-to: CloudFormation templates

The following instructions will assist you in deploying, configuring, and maintaining your AWS cloud infrastructure. The templates referenced within this guide take care of the deployment process of your AWS resources, so all that you will need to do is to perform the initial configuration and make updates to it as you see fit. This step-by-step guide involves both the use of the AWS Management Console, and the AWS CLI. Only one of these methods needs to be chosen to deploy your CloudFormation templates. Lastly, it is important to note that the individual user performing this setup must have permissions to create the resources described within this guide and the CloudFormation templates.

Configuring the infrastructure:

1. Upload the **vpc-config.yaml** and **rds-config.yaml** templates (in this order) to CloudFormation:
   a. AWS Management Console:
      i. Within the CloudFormation service, click **[Create stack]**-> **[With new resources (standard)]**-> **[Upload a template file]** -> **[Choose file]**.
      ii. **rds-config.yaml** will prompt you for database-specific parameters. Track this information for future use.
      iii. The rest of the configuration details are left up to (or you can simply continue through with the default settings).
   b. AWS CLI:

```
aws cloudformation create-stack --stack-name <name> --template-
body file://cloudformation/vpc-config.yaml
```

```
aws cloudformation create-stack --stack-name <name> --template-
body file://cloudformation/rds-config.yaml --parameters
ParameterKey=DBName,ParameterValue=<DB_Name>
ParameterKey=DBClass,ParameterValue=<DB_Class>
ParameterKey=DBAllocatedStorage,ParameterValue=<DB_Storage>
ParameterKey=DBUser,ParameterValue=<DB_Username>
ParameterKey=DBPassword,ParameterValue=<DB_Password>
```

Notes:

DBClass: db.t2.micro, db.t3.small, db.t3.medium or db.t3.large.
DBAllocatedStorage: 5 - 1024GB (enter the numerical value)

2. Upload the lambda function code to an S3 bucket.
    a. AWS Management Console:
        i. Under the S3 service, create an S3 bucket in the same AWS Region that you plan to run the CloudFormation templates. Be sure to track the name of the bucket for future use.
    b. AWS CLI:

```
aws s3api create-bucket --bucket <bucket_name> --region
<AWS_Region> --create-bucket-configuration
LocationConstraint=<AWS_Region>
```

    c. From the root project directory, add the necessary database parameters described in part 1 to the following files:
    d. /lambda/GetSurvey/db_config.py
    e. /lambda/PostResult/db_config.py
    f. /lambda/PostSurvey/db_config.py
    g. Then, upload the lambda function code to your S3 bucket by running this command from the root directory:

```
bash ./importLambda.sh <bucket_name>
```

3. Upload the **api-lambda-config.yaml** template to CloudFormation.
    a. AWS Management Console:
        i. Within the CloudFormation service, click **[Create stack]**-> **[With new resources (standard)]**-> **[Upload a template file]** -> **[Choose file]**.
        ii. This template will require approval for creating IAM roles on your behalf. This is done so that your resources have permissions to interact with each other.
        iii. The rest of the configuration details are left up to (or you can simply continue through with the default settings).
    b. AWS CLI:

```
aws cloudformation create-stack --stack-name <name> --template-
body file://cloudformation/api-lambda-config.yaml --capabilities
CAPABILITY_IAM CAPABILITY_NAMED_IAM --parameters
ParameterKey=S3BucketName,ParameterValue=<bucket_name>
```

Testing the infrastructure:

Now that the initial configuration of your resources has been completed, you can test to ensure that the correct connections have been established. Within the AWS Management Console, navigate to the API Gateway service and select the API Gateway named **fcniGateway**. Navigate to the GET method within the /survey resource and select test. Without adding any header information/parameters, click on test. Immediately following this, you should see the survey information appear on the screen. Now your resources have been properly configured to communicate with each other.

Notes:

1. Changes to the resources can be made programmatically, without needing to configure them within the AWS Management Console. This can be done by adding/removing parameters from resources within the CloudFormation templates and performing an update to the corresponding template that you have deployed.
    a. Make a copy of the original template and title it with the current version you're implementing.
    b. Reference the documentation to see what parameters you can add to give extra functionality to your resources, such as the RDS instance or VPC.
    c. Once you've completed your update, navigate to the CloudFormation service within the AWS Management Console and select the original template from your list of stacks then **[update]**. Or, you can perform this from the AWS CLI by executing the corresponding command with `create-stack` replaced with the keyword `update-stack`.
    d. On completion of this, the affected resources will be altered to fit your requirements. Note the possible downtime of certain resources, such as your RDS instance, during this update period.